



①9 BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENTAMT

⑫ Übersetzung der  
europäischen Patentschrift

⑤1 Int. Cl.<sup>6</sup>:  
**H 03 K 23/40**  
H 03 M 7/16

⑧7 EP 0 444 911 B1

⑩ **DE 691 13 836 T 2**

②1 Deutsches Aktenzeichen:	691 13 836.2
⑧6 Europäisches Aktenzeichen:	91 301 611.9
⑧6 Europäischer Anmeldetag:	27. 2. 91
⑧7 Erstveröffentlichung durch das EPA:	4. 9. 91
⑧7 Veröffentlichungstag der Patenterteilung beim EPA:	18. 10. 95
④7 Veröffentlichungstag im Patentblatt:	21. 3. 96

DE 691 13 836 T 2

③0 Unionspriorität: ③2 ③3 ③1  
01.03.90 US 487490

⑦3 Patentinhaber:  
Hewlett-Packard Co., Palo Alto, Calif., US

⑦4 Vertreter:  
Schoppe, F., Dipl.-Ing.Univ., Pat.-Anw., 82049 Pullach

⑧4 Benannte Vertragsstaaten:  
DE, GB

⑦2 Erfinder:  
Windmiller, Keith, Loveland, Colorado 80538, US

⑤4 Integrierter Hochgeschwindigkeitssynchronzähler mit asynchroner Auslesung.

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel III § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patentamt inhaltlich nicht geprüft.

DE 691 13 836 T 2

### Gebiet der Erfindung

Diese Erfindung bezieht sich auf digitale, elektronische Schaltungen und insbesondere auf digitale Zählschaltungen. Noch spezieller bezieht sich die Erfindung auf einen digitalen Hochgeschwindigkeitssynchronzähler mit der Fähigkeit zum asynchronen Auslesen.

### Hintergrund der Erfindung

Bei einer Messung der Zeit zwischen zwei Ereignissen haben herkömmliche Schaltungen das Auftreten des ersten Ereignisses verwendet, um einen Zähler zu starten, der Taktpulse zählt, und das Auftreten des zweiten Ereignisses, um den Zähler anzuhalten. Der Wert des Zählers wird dann gelesen, um die Zeit zwischen den Ereignissen zu bestimmen. Nach dem Lesen des Wertes des Zählers wird die Schaltung für die nächste Reihe von Ereignissen zurückgesetzt. Die Zeit, während der die Schaltung zurückgesetzt wird, wird Totzeit genannt, und während dieser Totzeit wird ein Auftreten der Ereignisse unbemerkt bleiben. Das Verhalten der Schaltung kann beträchtlich verbessert werden, wenn diese Totzeit beseitigt werden kann, wodurch ein Null-Totzeit-Zähler geschaffen wird. Ein Null-Totzeit-Zähler muß in der Lage sein, durchgehend zu zählen, und muß in der Lage sein, ein asynchrones Auslesen des Zählers zu ermöglichen, das das Zählen nicht stört. Beim asynchronen Auslesen der Ausgabe des Zählers tritt ein Metastabilitätsproblem auf, wenn ein Auslesen zu dem Zeitpunkt versucht wird, zu dem der Zähler einen Wert wechselt. Wenn die Ausgabe zwischengespeichert wird, während ein oder mehr Bits den Wert wechseln, könnten die wechselnden Bits entweder als eine Eins oder als eine Null zwischengespeichert werden, womit die Ausgabe metastabil ist. Dieses Problem ist bei Binärzählern sehr schwerwiegend, da viele

Stufen des Zählers gleichzeitig den Wert wechseln können, wenn beispielsweise ein Übertrag durch den Zähler durchgeschoben wird. Bei einem Binärzähler kann eine große Diskrepanz des Zählwerts auftreten, wenn die höherwertigen Bits zum Zeitpunkt des Auslesens wechseln.

Üblicherweise wurde dieses Metastabilitätsproblem auf eine von zwei Arten gelöst, wie von David C. Chu in seinem Artikel "Phase Digitizing: A New Method for Capturing and Analyzing Spread Spectrum Signals", Hewlett Packard Journal, Februar 1989, Seiten 28-33, beschrieben wird. Zwei Arten von Zählern können für diese Art von Schaltung verwendet werden. Ein herkömmlicher binärer Ripple-Zähler besteht aus einer Reihe von kaskadierten Halbiererstufen. Die maximale Zählrate wird tatsächlich durch die ersten wenigen Stufen bestimmt, wobei die Frequenz nach jeder Stufe halbiert wird. Nachfolgende Stufen beeinflussen die maximale Zählrate nicht, weswegen sie langsam und zahlreich sein können. Das Problem bei dieser Art von Zählern ist die Zeit, die benötigt wird, um einen Übertrag durch alle Stufen des Zählers durchzuschieben. Wenn die Ausgabe des Zählers gelesen wird, während gerade ein Übertrag-Bit durch den Zähler durchgeschoben wird, ist die Ausgabe metastabil und für alle Stufen, die den Wert wechseln, unbestimmt. Die herkömmliche Lösung dieses Problems besteht darin, genügend Zeit bereitzustellen, damit sich ein Übertrag durch alle Stufen ausbreiten kann, bevor ein Auslesen versucht wird. Dadurch wird die maximale Zählrate des Zählers schwerwiegend begrenzt.

Es kann auch ein synchroner Zähler verwendet werden, um das Metastabilitätsproblem zu lösen. Bei dem synchronen Zählen addiert ein logisches Netzwerk Eins zum gegenwärtigen Zählzustand hinzu, damit das nächste Ereignis den Zähler sofort auf den nächsten Zustand taktet. Bei einem sorgfältigem Entwurf kann der synchrone Zähler während des Durchlaufs gelesen werden, da alle Bits theoretisch gleichzeitig wechseln. Bei langen Zählketten wird das logische Netzwerk jedoch komplex und unhandlich, und die Verarbeitungszeit verlängert

sich. Da jedes Bit so schnell wie das schnellste Bit sein muß, kann der Leistungsverbrauch bei dieser Art von Schaltung erheblich sein.

Der Lösungsansatz von David Chu in dem Artikel, auf den im vorhergehenden Bezug genommen wurde, ist es, einen synchronen Zähler und einen Ripple-Zähler zu kombinieren und die Komplexität zu der Zwischenspeicherschaltung hinzuzufügen. Chu verwendet einen synchronen Zähler für die ersten wenigen Stufen, typischerweise acht Bit, und dann einen Ripple-Zähler für die restlichen Stufen. Zusätzlich sieht Chu eine Verzögerungsschaltung vor, um jedes Auslesen um eine kurze Zeit zu verzögern, nachdem ein Zählpuls in die Ripple-Stufen gesendet worden ist. Diese Verzögerung ist ausreichend lang, daß sich ein Übertrag durch alle Ripple-Stufen ausbreiten kann. Wenn ein Auslesesignal auftritt, werden die Bits des synchronen Abschnitts des Zählers ausgelesen und sofort zwischengespeichert, während die Bits des Ripple-Abschnitts des Zählers nach der Verzögerung, und somit nachdem sich der Übertrag ausgebreitet hat, ausgelesen werden.

Während dieser Lösungsansatz für bipolare Schaltungen gut funktioniert, ist er nicht für langsamere Niedrigleistungsprozesse, wie z.B. CMOS, geeignet. In der Technik besteht daher ein Bedarf nach einem synchronen Zähler, dessen Ausgaben asynchron zwischengespeichert werden können. Ferner besteht in der Technik ein Bedarf nach einem derartigen Zähler, der in einen CMOS-Prozess implementiert werden kann. Es besteht in der Technik noch ein zusätzlicher Bedarf nach einem solchen Zähler, der mit Niedrigleistungsschaltungen implementiert werden kann. Es besteht noch ein weiterer Bedarf nach einer solchen Schaltung mit einem vereinfachten Zwischenspeicherungsverfahren.

#### Zusammenfassung der Erfindung

Ein Aspekt der vorliegenden Erfindung ist es, einen Hochge-

schwindigkeitssynchronzähler mit der Fähigkeit zum asynchronen Auslesen zu schaffen.

Ein weiterer Aspekt der Erfindung ist es, einen derartigen Zähler zu schaffen, der das dem asynchronen Zwischenspeichern der Ausgabe des Zählers zugehörige Metastabilitätsproblem löst.

Ein weiterer Aspekt der Erfindung ist es, ein derartiges Metastabilitätsproblem zu lösen, indem ein Gray-Code-Zähler verwendet wird, bei dem bei jedem Zählen nur ein Bit wechselt, wodurch die zwischengespeicherte Ausgabe innerhalb eines Zählwerts des Zählers liegen wird.

Ein weiterer Aspekt der Erfindung ist es, ein Vorausschau-Qualifizierungssignal zu schaffen, um die Zählerinitialisierung des nächsten Zustands für obere Bits des Zählers von jedem Zyklus bis zu jedem achten Zyklus des Zählers zu strecken, wodurch der Frequenzbereich des Zählers erhöht wird.

Noch ein weiterer Aspekt der Erfindung ist es, die Eingabe in jedes Register der oberen Bits unter Verwendung einer seriellen Schaltung zu qualifizieren.

Die oben erwähnten und andere Aspekte der Erfindung werden in einem Hochgeschwindigkeitssynchronzähler erreicht, der einen Gray-Code zum Zählen verwendet. Da das Gray-Code-Zählen die einzigartige Eigenschaft besitzt, daß bei jedem neuen Zählwert nur ein einziges Bit des Zählers wechselt, kann die Ausgabe des Zählers zu jeder Zeit zwischengespeichert werden, und der zwischengespeicherte Wert wird niemals mehr als einen Zählwert von dem tatsächlichen Zählwert in dem Zähler entfernt sein. Im Gegensatz zu binären Zählern, bei denen eine Mehrzahl von Bits, manchmal sogar jedes Bit in dem Zähler, wechseln kann, wenn das Zählen den Zählerzustand von einem Zustand zu einem weiteren wechselt, wechselt ein Gray-Code-Zähler niemals mehr als ein Bit, wenn er von

einem Zustand zu einem weiteren zählt. Ferner hat dieser eine Bitwechsel nur einen Wechsel von Eins in dem Zählwert zur Folge, so daß der zwischengespeicherte Wert niemals mehr als einen Zählwert von dem Wert, der in dem Zähler enthalten ist, entfernt sein wird, wenn der Zähler zwischenspeichert wird, während dieses eine Bit gerade wechselt.

Die Verwendung eines Synchronzählers führt jedoch ein zusätzliches Problem in die Zählschaltung ein. Dieses Problem liegt darin, daß jedes Bit in dem Zähler die Werte aller Bits mit niedererem Wert durch eine logische UND-Verknüpfung verknüpfen muß, um zu bestimmen, wann das Bit gesetzt oder gelöscht werden soll. Daher benötigen bei einem großen Zähler die höherwertigen Bits UND-Gatter mit einer großen Anzahl von Eingängen. Das sechzehnte Bit eines 16-Bit-Zählers benötigt beispielsweise ein UND-Gatter mit 16 Eingängen, das achtundzwanzigste Bit eines 28-Bit-Zählers benötigt ein UND-Gatter mit 28 Eingängen, usw.. Dieses Problem wird bei der vorliegenden Erfindung dadurch gelöst, daß ein Vorausschau-Qualifizierungs-Bit geschaffen wird, das eine Vorausschau für acht Zählwerte schafft. Mit dieser Fähigkeit zur Vorausschau auf acht Zählwerte können die Qualifizierungseingaben in die höherwertigen Bits durch eine Reihe von UND-Gattern, von denen jedes nur zwei Eingänge besitzt, ausgeführt werden.

#### Kurze Beschreibung der Zeichnungen

Die obigen und weitere Aspekte, Merkmale und Vorteile der Erfindung werden durch das Lesen der folgenden, detaillierteren Beschreibung der Erfindung, die in Verbindung mit den folgenden Zeichnungen dargelegt wird, offensichtlicher. Es zeigen:

Fig. 1 ein Blockdiagramm der Erfindung;

Fig. 2 ein logisches Diagramm, das die vier niederwertigen

Bits des Zählers zeigt und das obere AUX-Signal darstellt;

- Fig. 3 ein Zeitdiagramm des Betriebs der Erfindung;
- Fig. 4 ein logisches Diagramm, das die nächsthöheren vier Bits des Zählers zeigt und die Verwendung der seriellen UND-Schaltung darstellt; und
- Fig. 5 ein Blockdiagramm, das das Verfahren und die Vorrichtung zur Umwandlung des Gray-Codes in einen Binär-Code zeigt.

#### Beschreibung des bevorzugten Ausführungsbeispiels

Die folgende Beschreibung zeigt die Art und Weise der Ausführung der vorliegenden Erfindung, die gegenwärtig als die beste angesehen wird. Diese Beschreibung soll nicht im Sinne einer Begrenzung verstanden werden, sondern sie wird nur zum Nutzen der Beschreibung der allgemeinen Prinzipien der Erfindung durchgeführt. Der Bereich der Erfindung soll Bezugnehmend auf die beigefügten Ansprüche bestimmt werden.

Fig. 1 zeigt ein Blockdiagramm der Erfindung. Wie in Fig. 1 gezeigt ist, wird Ein Gray-Code-Zähler 102 von einem Takt 110 getrieben. Die Ausgabe des Gray-Code-Zählers 102 wird zu einem Latch 104 übertragen, wenn ein Latch-Signal 112 aktiviert wird. Die Ausgabe des Latch 104 wird durch den Gray-Code/Binär-Wandler 106 von dem Gray-Code in einen Binär-Code umgewandelt. Der Gray-Code-Zähler 102 besitzt bei der vorliegenden Erfindung typischerweise 28 Bits, wobei die Erfindung jedoch nicht auf irgendeine besondere Länge des Zählers begrenzt ist. Sie könnte auch für Zähler verwendet werden, die bedeutend größer oder kleiner als 28 Bits sind.

Der Gray-Code besitzt bei Hochgeschwindigkeitszählern einen bedeutenden Vorteil, da er die Eigenschaft besitzt, daß in

dem gesamten Zähler lediglich ein Bit wechselt, wenn er von einem Zustand zu dem nächsten zählt. Dieser Aspekt ist besonders wichtig, wenn die Ausgabe des Zählers zwischengespeichert werden muß. Wegen der Eigenschaft, daß nur ein Bit in dem Zähler von einem Zustand zum nächsten wechselt, wenn die Ausgabe des Zählers während des Auftretens eines Zustandswechsels zwischengespeichert wird, wird sich die Ausgabe niemals mehr als einen Zählwert von dem Wert des Zählers unterscheiden. Wenn beispielsweise bei einem 4-Bit-Zähler ein binäres Zählverfahren verwendet werden würde, würde es möglich sein, eine Ausgabe von fünfzehn zu erzeugen, wenn der Zähler vom Zustand 7 zum Zustand 8 wechselt. Wenn das hochwertigste Bit schneller als die drei niederwertigen Bits von einer Eins zu einer Null wechselt, könnte die Ausgabe des Zählers als lauter Einsen gelesen werden, was einem Zustand von 15 entspricht. Dieser Sachverhalt ist bei einem Gray-Code-Zähler nicht möglich.

Eine einfache Regel zum Erzeugen von Gray-Code-Zuständen ist es, mit dem Zustand von lauter Nullen zu beginnen. Um zu einem nächsten Zustand zu wechseln, wird immer das einzelne niederstwertigste Bit gewechselt, was einen vorher unbenutzten Zustand zur Folge haben wird. Tabelle 1 zeigt die Zustände der letzten sechs Bits des Zählers der vorliegenden Erfindung für die ersten 33 Zählwerte. Die Betriebsweise eines Gray-Code-Zählers kann einfach verstanden werden, indem die Bits Q24, Q25, Q26 und Q27 für die Zeilen betrachtet werden, die Dezimalwerte von null bis fünfzehn besitzen. In Zeile 0 sind alle vier Bits, Q24 bis Q27, Null. Wenn der Zählwert auf Eins inkrementiert wird, wechselt Q27 auf eine Eins, was einen Zustand von 0001 zur Folge hat. Wenn der Zählwert auf zwei inkrementiert wird, wechselt Q26 auf eine Eins, was einen Zustand von 0011 zur Folge hat. Wenn der Zählwert auf drei inkrementiert wird, wechselt Q27 zurück auf Null, was einen Zustand von 0010 zur Folge hat. Dies setzt sich bis zu einem Zählwert von 15 fort, der einem Zustand von 1000 entspricht. Es ist zu beachten, daß bei jedem Zustandswechsel mit jedem neuen Zählwert nur eines der vier Bits wechselt.



Ein einfacherer Weg um die Zustände elektronisch zu erzeugen ist es, ein niederwertiges Bit AUX zu schaffen. Das LSB AUX (LSB = Least Significant Bit = niederwertiges Bit) wird geschaffen, indem die Takteingabe des Zählers durch zwei geteilt wird. Wenn das LSB AUX verwendet wird, können die Zustände des Zählers unter Verwendung der folgenden Regel geschaffen werden:

Ein Bit des Zählers wird bei dem nächsten Taktsignal umgeschaltet, wenn:

- A. das nächste niederwertige Bit Eins ist,
- B. und alle anderen niederwertigen Bits Null sind,
- C. und das LSB AUX Eins ist.

Die einzige Ausnahme dieser Regel ist das niederwertige Bit des Zählers, in diesem Beispiel Q27, das immer dann umgeschaltet wird, wenn das Bit LSB AUX Null ist. Dieses Zählverfahren kann leicht aus dem Nachvollziehen einiger Zeilen von Tabelle 1 und den Werten des Bits LSB AUX gesehen werden.

Fig. 2 zeigt ein schematisches Diagramm der Bits Q24 bis Q27 des Zählers der vorliegenden Erfindung. Tabelle 2 zeigt die logischen Gleichungen, um die Eingaben in die Flip-Flops aus Fig. 2 zu erzeugen, wobei "\*" ein logisches UND und "~" eine Inversion darstellen. Bezugnehmend nun auf Fig. 2 und Tabelle 2 speichern D-Flip-Flops 202, 204, 206 und 208 die vier niederwertigen Bits des Zählers 102 (Fig. 1) der vorliegenden Erfindung. Das Bit LSB AUX 210 wird von einem Invertierer 214 invertiert und in das Flip-Flop 202 des niederwertigen Bits 27 eingegeben. Die logische Gleichung für diese Eingabe ist in Tabelle 2, Zeile 1 gezeigt. Der invertierte Flip-Flop-Ausgang 224 des Bits 27 ist mit einem UND-Gatter verbunden, dessen andere Eingabe das Bit LSB AUX 210 ist.

Der Ausgang des UND-Gatters 216 liefert die Eingabe in das D-Flip-Flop 204 des Bits 26 des Zählers. Die logische Gleichung für diese Eingabe ist in Zeile 2 von Tabelle 2 gezeigt, die zeigt, daß die Eingabe die UND-Verknüpfung des Bits LSB AUX und des Komplements des Bits 27 ist. Wie in Zeile 3 der Tabelle 2 dargestellt ist, ist die Eingabe in das D-Flip-Flop 206 des Bits 25 die logische UND-Verknüpfung des Bits LSB AUX der invertierten Ausgabe des Bit 27 und der nicht invertierten Ausgabe des Bit 26. Die Eingabe in ein D-Flip-Flop 208 des Bit 24 ist die logische UND-Verknüpfung des Bits LSB AUX, der invertierten Ausgabe des Bits 27, der invertierten Ausgabe des Bits 26 und der nicht invertierten Ausgabe des Bits 25.

Wenn die Konstruktion der anderen Bits des 28-Bit-Zählers der Erfindung in einer ähnlichen Weise durchgeführt wird, benötigt die UND-Gatter-Eingabe in jedes höherwertige Bit ein zusätzliches Eingangssignal, wie durch die obige Beschreibung von Fig. 2 gezeigt ist. Das logische Ergebnis ist, daß das Bit Null, das höchstwertige Bit, des Zählers eine UND-Funktion mit 28 Eingängen benötigt. Wenn eine solche Funktion in einer integrierten Schaltung implementiert werden würde, würde sie einen beträchtlichen Flächenbetrag der integrierten Schaltung benötigen und einen enormen Leistungsbetrag verbrauchen. Um den verbrauchten Leistungsbetrag und den Flächenbetrag der integrierten Schaltung zu reduzieren, schafft die vorliegende Erfindung ein Bit UPPER AUX, das dazu verwendet wird, alle Bits von höherer Ordnung als die vier niederwertigen Bits des Zählers zu qualifizieren. Dieses Bit UPPER AUX wird es ermöglichen, daß die Zählerinitialisierung des "nächsten Zustandes" von jedem Zyklus bis zu jedem achten Zyklus gestreckt wird. Wegen dieser zusätzlichen Zeit zum Initialisieren der höherwertigen Bits kann eine andere Schaltung verwendet werden, die eine beträchtliche Leistung und Fläche der integrierten Schaltung einspart. Das Bit UPPER AUX ist definiert als:

$$\text{LSB AUX} * \sim Q27 * \sim Q26 * \sim Q25$$

Ein UND-Gatter 222 in Fig. 2 erzeugt das UPPER AUX Signal 240 der vorliegenden Erfindung.

Tabelle 3 stellt die logischen Gleichungen für die Eingaben in die Flip-Flops dar, wobei das Bit UPPER AUX dazu verwendet wird, die Eingaben für die höherwertigen Bits Q23, Q22, etc., zu erzeugen.

Fig. 3 zeigt ein Zeitablaufdiagramm der Schaltung, das die Takteingabe, das Signal LSB AUX, das Signal UPPER AUX und die Ausgaben der sechs niederwertigen Flip-Flops des Zählers aufweist. Bezugnehmend nun auf Fig. 3 stellt das Zeitablaufdiagramm dar, daß das Flip-Flop Q24 acht Zyklen vor dem Wechsel des Bits UPPER AUX in einen hohen Zustand in einen hohen Zustand wechselt. Das Diagramm zeigt ferner, daß die höherwertigen Bits nur umschalten, nachdem das Bit UPPER AUX einen hohen Zustand besitzt. Wegen den acht Taktzyklen zwischen dem Wechsel von Q24 auf einen hohen Zustand und dem Wechsel des Bits UPPER AUX auf einen hohen Zustand haben die Eingaben in die höherwertigen Bits eine Zeit von acht Taktzyklen, um einzuschwingen. Diese Einschwingzeit ist lang genug, daß statt der parallelen UND-Funktionen, die für die vier niederwertigen Bits benötigt werden, serielle UND-Funktionen verwendet werden können. Diese seriellen UND-Funktionen schaffen, wie hierin nachfolgend dargelegt wird, bedeutende Vorteile im Leistungs- und Platzverbrauch der integrierten Schaltung.

Fig. 4 ist ein schematisches Diagramm der nächsthöheren vier Bits, der Bits 23 bis 20, des Zählers der vorliegenden Erfindung. Bezugnehmend nun auf Fig. 4, stellen D-Flip-Flops 402, 404, 406 und 408 die Bits 23, 22, 21, bzw. 20 des Zählers dar. Ein UND-Gatter 410 qualifiziert die Eingabe in ein Flip-Flop 402 des Bits 23. Wie auch in Tabelle 3 gezeigt ist, ist die Eingabe in Bit 23 eine logische UND-Verknüpfung des Bits UPPER AUX und der nicht invertierten Ausgabe von Q24. Das Signal QUAL IN 450 für dieses Bit ist eine logische

Eins, die typischerweise an VDD gebunden ist. Wegen der Verwendung des Bits UPPER AUX und der Verwendung der seriellen UND-Funktionen besitzen alle anderen qualifizierenden UND-Gatter zu allen anderen Flip-Flops des Zählers drei Signaleingänge, wie hierin nachfolgend beschrieben wird. Dies ist durch ein UND-Gatter 412 dargelegt, das die Eingabe in das Bit 22 ist. Die Eingaben in das UND-Gatter 412 umfassen das Bit UPPER AUX 240, die nicht invertierte Ausgabe 426 des Bits 23 und die invertierte Ausgabe 234 des Bits 24 auf. Die qualifizierende Eingabe in das Bit 21 wird von einem UND-Gatter 414 erzeugt, dessen Eingaben das Bit UPPER AUX 240, die nicht invertierte Ausgabe 430 des Bits 22 und die logische UND-Verknüpfung aus der invertierten Ausgabe 428 des Bits 23 und der invertierten Ausgabe 234 des Bits 24 sind. Ein UND-Gatter 418 ist das erste der seriellen UND-Gatter, die dazu verwendet werden, qualifizierende Eingaben in die oberen Flip-Flops zu erzeugen. Wie in Tabelle 3 gezeigt ist, wird die qualifizierende Eingabe für die Flip-Flops Q21 und höher immer das Bit UPPER AUX, die nicht invertierte Ausgabe des nächstniedrigeren Flip-Flops und die logische UND-Verknüpfung aus den invertierten Ausgaben aller anderen niedrigeren Flip-Flops bis herunter zu Q24 aufweisen. Diese logische UND-Verknüpfung der invertierten Ausgänge der niederen Flip-Flops wird von den seriellen UND-Gattern 418, 420, 422 und 424 gebildet. Jedes der seriellen UND-Gatter 418, 420, usw., bildet die logische UND-Verknüpfung des invertierten Ausgangs eines Flip-Flops, das mit den invertierten Ausgängen aller vorherigen Flip-Flops über die logische UND-Verknüpfung verbunden ist. Diese seriellen UND-Gatter sparen beträchtlich Leistung und beträchtlich Platz auf der integrierten Schaltung, wobei sie es immer noch ermöglichen, daß der Zähler als ein Synchronzähler arbeitet. Für die Bits 19 und höher wird die Schaltung von 4 reproduziert, wobei das Signal QUAL OUT 446 der nächstniedrigeren vier Bits mit dem Signal QUAL IN 450 verbunden ist, das Signal 440 des Bits 20 der nächstniedrigeren vier Bits mit dem Signal 236 des Bits 24, und das Signal 448 des -Bit 20 der nächstniedrigeren vier Bits mit dem Signal 234 des -Bit 24 verbunden ist.

Wie im vorhergehenden beschrieben ist, schafft der Gray-Code-Zähler 102 (Fig. 1) der vorliegenden Erfindung ein Verfahren zum Zählen, bei dem bei jedem neuen Zählwert nur ein Bit wechselt. Die meisten anderen Schaltungen benötigen jedoch eine Zählerausgabe, die eine binäre Form aufweist. Da der Gray-Code-Zähler bei den meisten Schaltungen nützlich ist, muß seine Ausgabe von dem Gray-Code in einen Binär-Code umgewandelt werden. Fig. 5 zeigt ein Blockdiagramm der Latch-Ausgabe des Gray-Code-Zählers und die Gray-Code/Binär-Wandler-Schaltungen. Bezugnehmend nun auf Fig. 5 sind zwölf Bits der Ausgabe des Gray-Code-Zählers 102 (Fig. 1) gezeigt. Diese zwölf Bits werden mit drei 4-Bit-Latches 502, 504 und 506 verbunden. Diese 4-Bit-Latches 502, 504 und 506 sind ein Teil der Latch-Schaltung 104 (Fig. 1). Die 4-Bit-Latches 502, 504 und 506 speichern die Ausgabe des Gray-Code-Zählers 102 zwischen, wenn sie das Latch-Signal 112 empfangen. Wegen der Hochgeschwindigkeitseigenschaft des Gray-Code-Zählers der vorliegenden Erfindung ist es auch sehr wichtig, daß die Gray-Code/Binär-Wandlung bei hoher Geschwindigkeit durchgeführt wird. Eine Gray-Code/Binär-Wandlung wird normalerweise durch eine EXKLUSIV-ODER-Verknüpfung des Ausgangs jedes Bits des Gray-Codes mit dem umgewandelten Bit nächsthöherer Ordnung erreicht. Das höchstwertige Bit (MSB; MSB = Most Significant Bit) ist jedoch bei dem Gray-Code und dem Binär-Code gleich. Daher kann das höchstwertige Bit der Ausgabe des Gray-Code-Zählers direkt in die umgewandelte Binärausgabe kopiert werden. Durch den Ripple-Effekt dieser Art von Umwandlung wird viel Zeit verloren. Diese Umwandlung wird durch die logischen Gleichungen von Tabelle 4 dargestellt, wobei "@" die EXKLUSIV-ODER-Funktion darstellt.

Das Verhalten der Gray-Code/Binär-Code-Wandlung wird durch die Aufteilung des Zählers in 4-Bit-Abschnitte verbessert. Jeder 4-Bit-Abschnitt wird anfangs auf die vorher beschriebene Weise von dem Gray-Code in den Binär-Code umgewandelt, als ob er ein isolierter 4-Bit-Zähler wäre. Das heißt, daß

während der anfänglichen Umwandlung jeder vier Bits die Werte aller höherwertigen Bits ignoriert werden. Nach der anfänglichen Umwandlung wird eine abschließende Umwandlungsstufe durchgeführt, indem die niederwertigen Bits jedes höherwertigen 4-Bit-Abschnitts untersucht werden. Diese niederwertigen Bits werden alle durch eine EXKLUSIV-ODER-Verknüpfung verknüpft, wobei, wenn das Resultat eine Eins ist, alle Bits des 4-Bit-Abschnittes invertiert werden.

Fig. 5 stellt schematisch diesen Umwandlungsprozeß dar. Jeder 4-Bit-Gray/Binär-Wandler 508, 510 und 512 wandelt die vier Bits der Gray-Code-Zählerausgabe in eine 4-Bit-Binär-Ausgabe um, während er alle höherwertigen Bits ignoriert. Die 4-Bit-EXKLUSIV-ODER-Schaltung 518 verknüpft jeden der 4-Bit-Ausgänge des Gray/Binär-Wandlers 512 mit dem Ausgang der EXKLUSIV-ODER-Schaltung 520 mit einer EXKLUSIV-ODER-Verknüpfung. Die EXKLUSIV-ODER-Schaltung 520 untersucht das niederwertige Bit 530 der Ausgabe des Gray-Binär-Wandlers 510 mit den aus einer EXKLUSIV-ODER-Verknüpfung aller niederwertigen Bits aller Wandler der höheren Abschnitte entstandenen Ergebnissen. Wenn die Ausgabe des EXKLUSIV-ODER-Gatters 520 eine logische Eins ist, werden alle vier Bits, die von dem Gray/Binär-Wandler 512 ausgegeben werden, durch die 4-Bit-EXKLUSIV-ODER-Schaltung 518 invertiert. Auf ähnliche Weise wird die EXKLUSIV-ODER-Schaltung 516 die vier Bits der Ausgabe des Gray/Binär-Wandlers 510 invertieren, wenn die Ausgabe eines EXKLUSIV-ODER-Gatters 522 eine logische Eins darstellt. Auf diese Weise wandeln die seriellen EXKLUSIV-ODER-Gatter 520, 522, usw., zusammen mit dem Zweistufenumwandlungsverfahren die Gray-Code-Ausgabe des Zählers mit einer hoher Geschwindigkeit in einen Binärwert um.

Nachdem nun ein derzeit bevorzugtes Ausführungsbeispiel der vorliegenden Erfindung beschrieben worden ist, wird nun verdeutlicht, daß die Aspekte der Erfindung vollständig erreicht worden sind, und es wird Fachleuten offensichtlich sein, daß viele Änderungen in Konstruktion und Schaltungsaufbau und weit abweichende Ausführungsbeispiele und Anwen-

dungen der Erfindung möglich sind, ohne vom Bereich der vorliegenden Ansprüche abzuweichen. Die Offenbarungen und die hierin enthaltene Beschreibung sind dazu bestimmt, anschaulich zu sein, und sie begrenzen in keiner Weise die Erfindung, deren Schutzbereich durch die folgenden Ansprüche definiert ist.

TABELLE 1

Gray-Code-Zähler

<Die niederstwertigen sieben Bits>

<u>MSB</u>						<u>LSB</u>				<u>LSB</u>	
<u>Q22</u>	<u>Q23</u>	<u>Q24</u>	<u>Q25</u>	<u>Q26</u>	<u>Q27</u>	<u>HEX</u>	<u>DEZ</u>	<u>AUX</u>			
0	0	0	0	0	0	00	0	0			
0	0	0	0	0	1	01	1	1			
0	0	0	0	1	1	02	2	0			
0	0	0	0	1	0	03	3	1			
0	0	0	1	1	0	04	4	0			
0	0	0	1	1	1	05	5	1			
0	0	0	1	0	1	06	6	0			
0	0	0	1	0	0	07	7	1			
0	0	1	1	0	0	08	8	0			
0	0	1	1	0	1	09	9	1			
0	0	1	1	1	1	0a	10	0			
0	0	1	1	1	0	0b	11	1			
0	0	1	0	1	0	0c	12	0			
0	0	1	0	1	1	0d	13	1			
0	0	1	0	0	1	0e	14	0			
0	0	1	0	0	0	0f	15	1			
0	1	1	0	0	0	10	16	0			
0	1	1	0	0	1	11	17	1			
0	1	1	0	1	1	12	18	0			
0	1	1	0	1	0	13	19	1			
0	1	1	1	1	0	14	20	0			
0	1	1	1	1	1	15	21	1			
0	1	1	1	0	1	16	22	0			
0	1	1	1	0	0	17	23	1			
0	1	0	1	0	0	18	24	0			
0	1	0	1	0	1	19	25	1			
0	1	0	1	1	1	1a	26	0			
0	1	0	1	1	0	1b	27	1			
0	1	0	0	1	0	1c	28	0			
0	1	0	0	1	1	1d	29	1			



0	1	0	0	0	1	1e	30	0
0	1	0	0	0	0	1f	31	1
1	1	0	0	0	0	20	32	0
1	1	0	0	0	1	21	33	1

TABELLE 2

Zeile Bit Qualifizierung (Umschalten beim nächsten Takt,  
wenn 1)

1	Q27	~LSB_AUX	*				
2	Q26	LSB_AUX	*	~Q27			
3	Q25	LSB_AUX	*	~Q27	*	Q26	
4	Q24	LSB_AUX	*	~Q27	*	~Q26	* Q25
.	.	.	.	.	.	.	.

TABELLE 3

Zeile Bit Qualifizieren (Umschalten beim nächsten Takt,  
wenn 1)

1	Q27	~LSB_AUX					
2	Q26	LSB_AUX	*	~Q27			
3	Q25	LSB_AUX	*	~Q27	*	Q26	
4	Q24	LSB_AUX	*	~Q27	*	~Q26	* Q25
5	Q23	UPPER_AUX	*	Q24			
6	Q22	UPPER_AUX	*	~Q24	*	Q23	
7	Q21	UPPER_AUX	*	~Q24	*	~Q23	* Q22
8	Q20	UPPER_AUX	*	~Q24	*	~Q23	* ~Q22 * Q21
9	Q19	UPPER_AUX	*	~Q24	*	~Q23	* ~Q22 * ~Q21 * Q20
.	.	.	.	.	.	.	.

**TABELLE 4**

Umwandlung vom Gray-Code in den Binär-Code

b0	(msb)	=	g0	(msb)
b1		=	g1	⊕ b0
b2		=	g2	⊕ b1
b3		=	g3	⊕ b2
b4		=	g4	⊕ b3

•  
•  
•

Europäisches Aktenzeichen: 91301611.9-2304

Patentansprüche

1. Eine digitale Hochgeschwindigkeitssynchronzählerschaltung (102) mit der Fähigkeit zum asynchronen Auslesen, wobei der Zähler folgende Merkmale aufweist:  
  
eine Mehrzahl von Einrichtungen zum Speichern niedrigerer Bits (202, 204, 206, 208);  
  
eine Mehrzahl von Einrichtungen zum Speichern höherer Bits (402, 404, 406, 408);  
  
eine Einrichtung zum Bewirken, daß die Einrichtungen zum Speichern niedrigerer Bits und höherer Bits unter Verwendung eines Gray-Codes zählen (214, 216, 218, 220, 410, 412, 414, 416); und  
  
eine Mehrfach-Zählwert-Vorausschau-Qualifizierungseinrichtung (222), um einen Zählerüberlauf von den Einrichtungen zum Speichern niedrigerer Bits zu den Einrichtungen zum Speichern höherer Bits zu übertragen.
2. Die digitale Zählerschaltung gemäß Anspruch 1, die ferner eine serielle Qualifizierungseinrichtung für jede der Mehrzahl von Einrichtungen zum Speichern höherer Bits (418, 420, 422, 424) aufweist.
3. Die digitale Zählerschaltung gemäß Anspruch 1, bei der die Mehrfach-Zählwert-Vorausschau-Qualifizierungseinrichtung (222) eine Einrichtung zum Schaffen einer Vorausschau auf acht Zählwerte aufweist.
4. Der digitale Zähler gemäß Anspruch 1, der ferner eine asynchrone Zwischenspeichereinrichtung (104) zum

Zwischenspeichern einer Ausgabe des digitalen Zählers aufweist.

5. Der digitale Zähler gemäß Anspruch 1, bei dem die Mehrzahl der Einrichtungen zum Speichern niedrigerer Bits vier Bits aufweist.

6. Ein Verfahren zum schnellen, synchronen und digitalen Zählen, während ein asynchrones Auslesen eines Zählwerts ermöglicht ist, mit folgenden Schritten:

Vorsehen eines Zählers mit Zählregistern (202, 204, 206, 208) für niedrigere Bits und Zählregistern (402, 404, 406, 408) für höhere Bits;

Wechseln eines Zustands des Zählers gemäß einem Gray-Code, wenn ein Takt empfangen wird; und

Qualifizieren der Zählregister für höhere Bits eine Mehrzahl von Zählwerten vor einem Zählwertüberlauf der Zählregister für niedrigere Bits (222).

7. Das Verfahren gemäß Anspruch 6, bei dem der Schritt des Qualifizierens ferner den Schritt des seriellen Qualifizierens der Register (418, 420, 422, 424) acht Zählwerte vor dem Zählwertüberlauf (222) aufweist.

Europäisches Aktenzeichen 91301611.9-2304

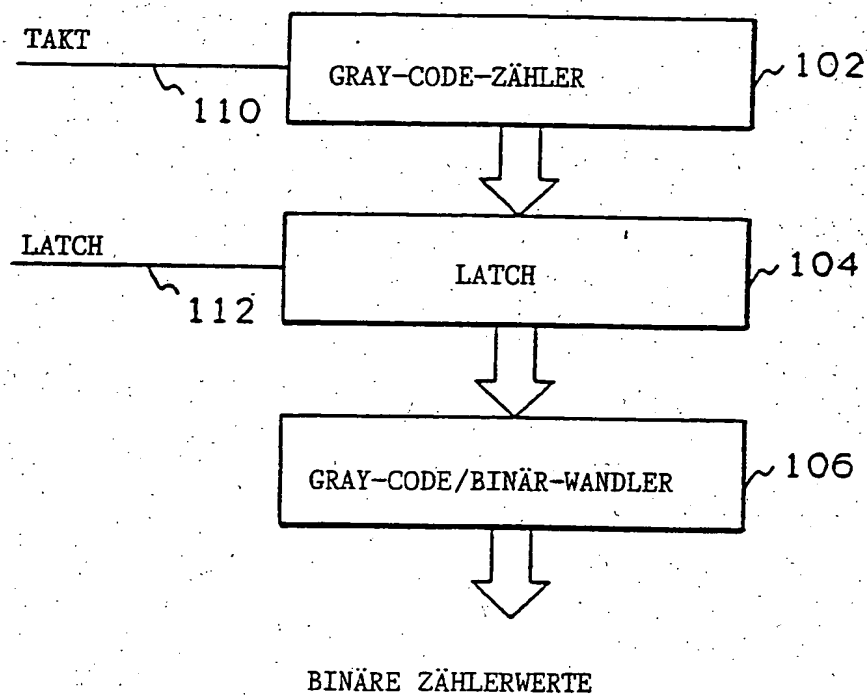


FIG. 1

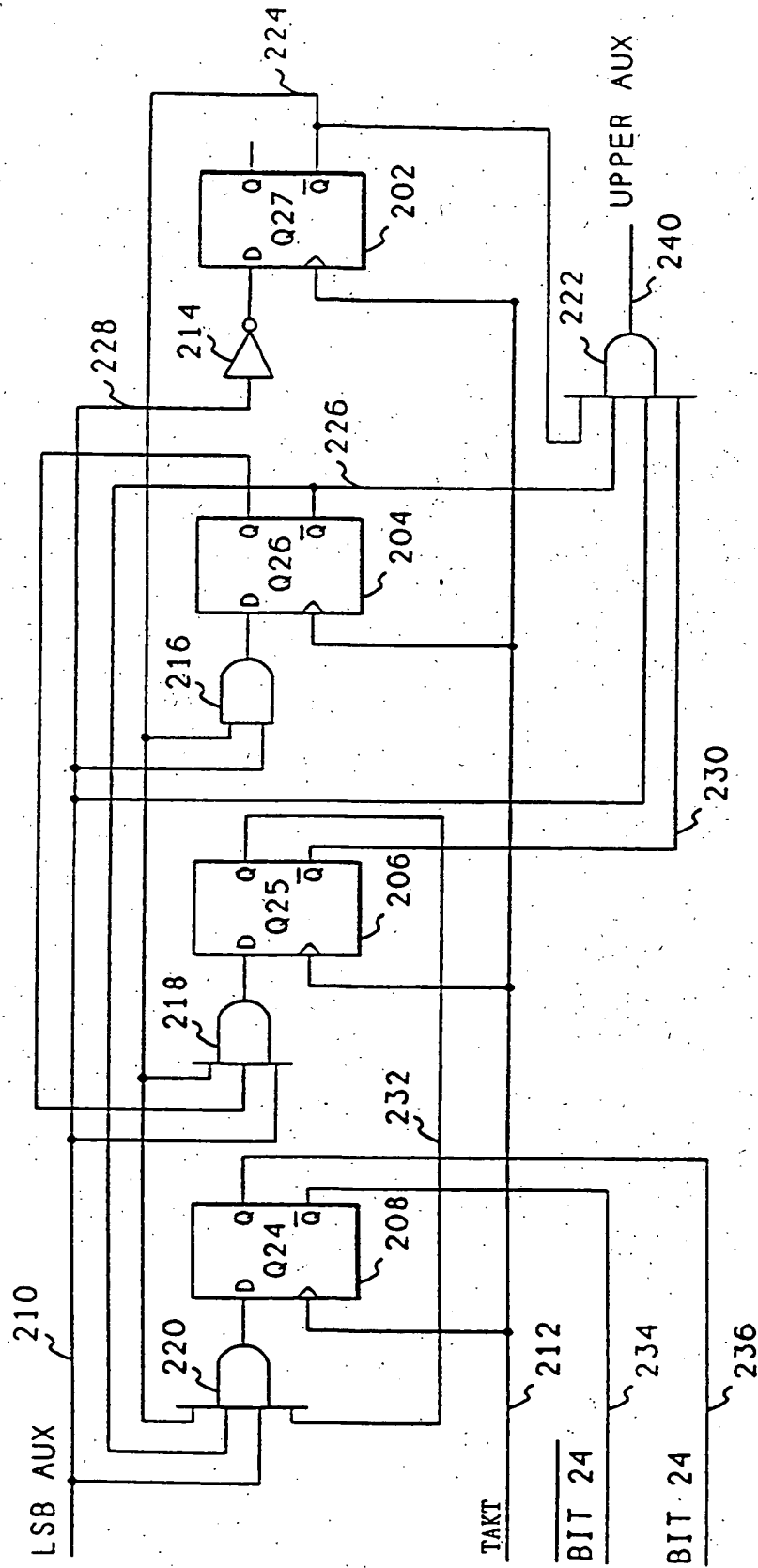


FIG. 2

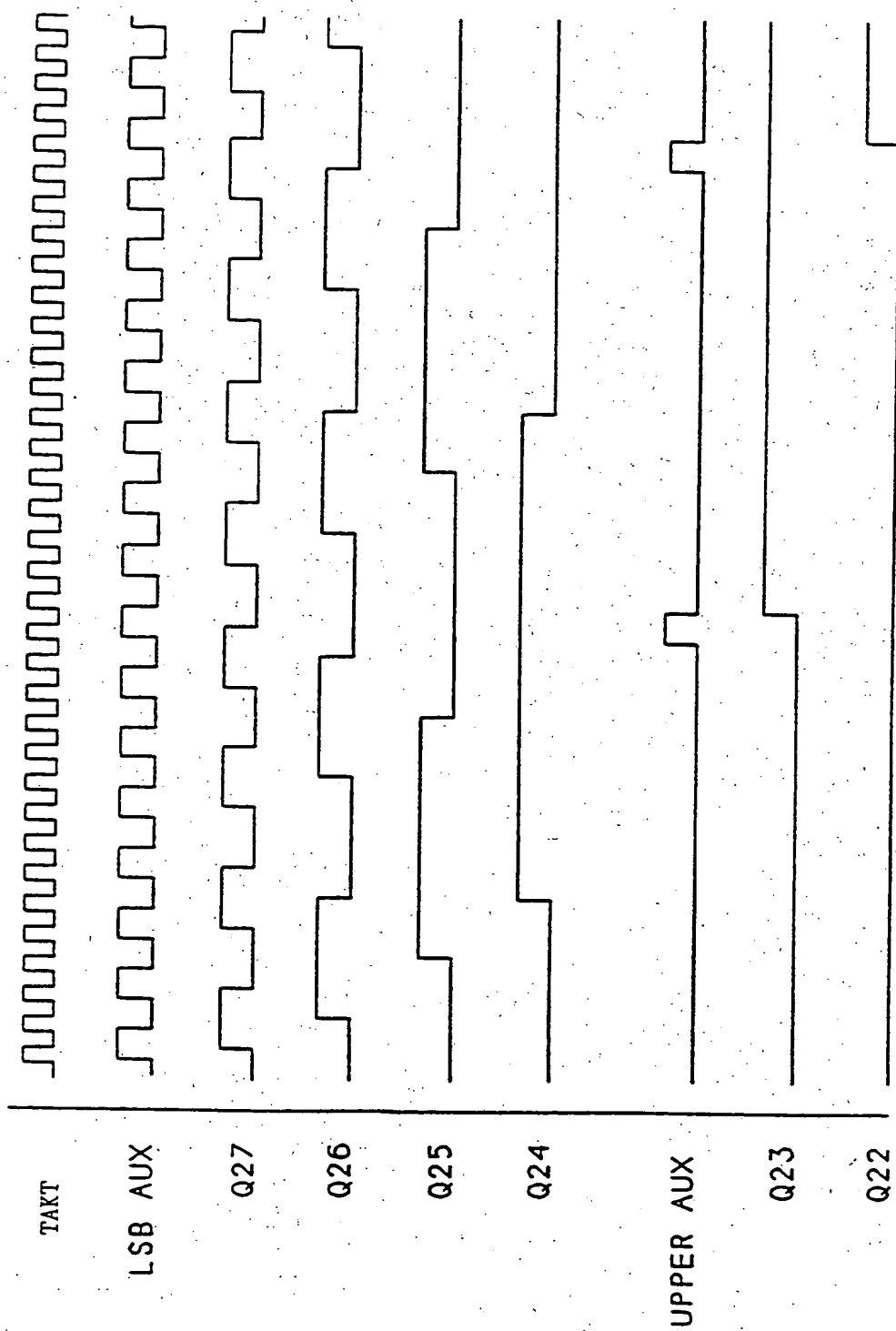


FIG. 3

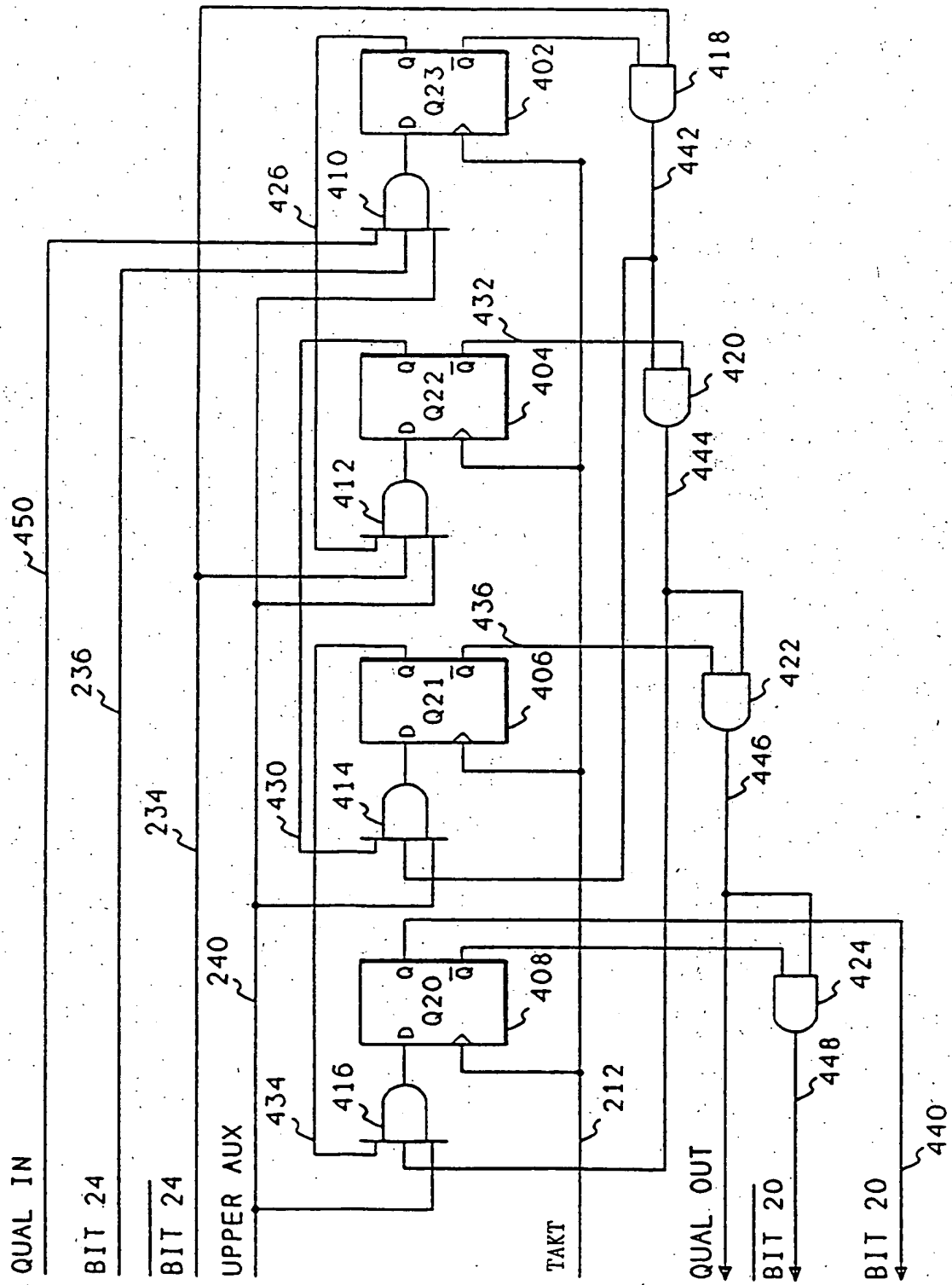


FIG. 4



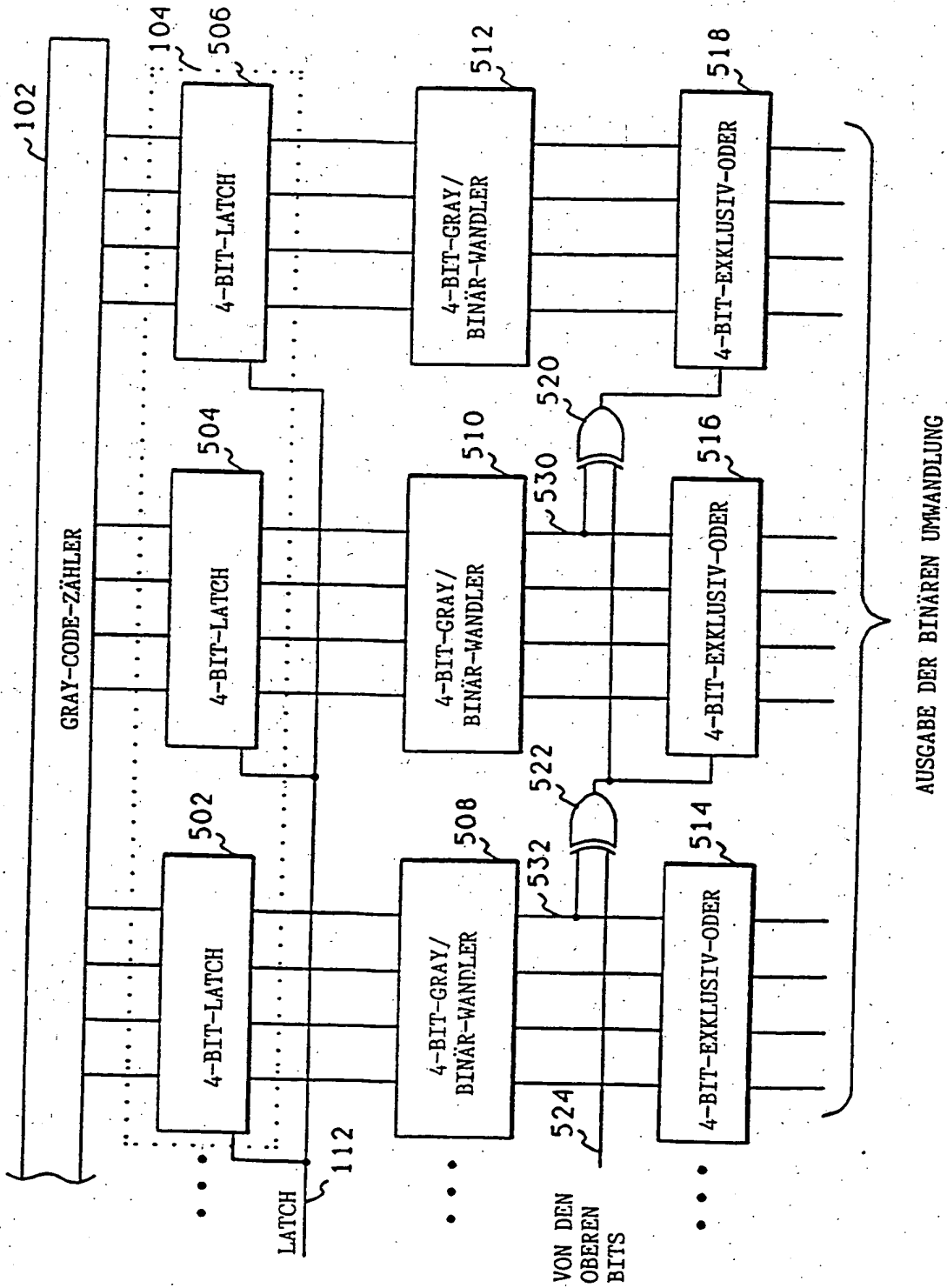


FIG. 5

